

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) A machine automation system for automating control of software tests on a client machine under control of a server process, the system comprising:

a predefined machine automation server object class adapted to execute in the server process;

a predefined machine automation client object class adapted to execute on the client machine in communication with an instance of the machine automation server object class, and the predefined machine automation client object class comprises one or more machine automation client objects for executing testing methods on the client machine; and

a machine automation control module instantiating a machine automation server object of the machine automation server object class in the server process and instructing the machine automation server object to instantiate one of the machine automation client objects of the machine automation client object class on the client machine to control the testing of the client machine.

2. (Previously Presented) The machine automation system of claim 1 further comprising:

a machine identifier identifying the client machine on which the machine automation client object is to be instantiated; and

a machine identifier source providing the machine identifier to the machine automation server object to initiate instantiation of the machine automation client object on the client machine.

3. (Original) The machine automation system of claim 1 wherein the machine automation server object includes a shutdown server object executing in the server process of a server machine, the machine automation client object includes a shutdown client object executing in a client process of the client machine, and the machine automation control module instructs the shutdown server object to cause the shutdown client object to reboot the client machine and to re-establish communications with the shutdown client object via a communications mechanism after rebooting of the client machine completes.

4. (Original) The machine automation system of claim 3 wherein the machine automation control module continues execution after rebooting of the client machine completes.

5. (Original) The machine automation system of claim 4 further comprising:
a timeout field of the machine automation server object for storing a timeout value specifying an amount of time the machine automation server object waits to return control to the machine automation control module after rebooting of the client machine completes.

6. (Original) The machine automation system of claim 3 wherein the machine automation control module continues execution after the shutdown server object causes the shutdown client object to reboot the client machine and before rebooting of the client machine completes.

7. (Original) The machine automation system of claim 1 wherein the machine automation server object includes a shutdown server object executing in the server process of a

server machine, the machine automation client object includes a shutdown client object executing in a client process of the client machine, and the machine automation control module instructs the shutdown server object to cause the shutdown client object to log off of the client machine and to re-log in to the client machine using a predetermined user name without rebooting.

8. (Original) The machine automation system of claim 7 wherein the predetermined user name is recorded in a system registry of the client machine and read from the system registry by the machine automation client object to re-log in to the client machine.

9. (Original) The machine automation system of claim 1 wherein the machine automation server object includes a shutdown server object executing in the server process of a server machine, the machine automation client object includes a shutdown client object executing in a client process of the client machine, and the machine automation control module instructs the shutdown server object to cause the shutdown client object to reboot the client machine and to re-log in to the client machine after rebooting completes.

10. (Original) The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to access a system registry of the client machine.

11. (Original) The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to return system information about the client machine to the machine automation server object.

12. (Original) The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to return to the machine automation server object a snapshot of a system registry of the client machine.

13. (Original) The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to return to the machine automation server object a snapshot of a file system of the client machine. The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to install an application on the client machine.

14. (Original) The machine automation system of claim 1 wherein the machine automation control module instructs the machine automation server object to cause the machine automation client object to install an application on the client machine.

15. (Previously Presented) The machine automation system of claim 1 further comprising:

a first machine identifier received by the machine automation server object identifying the client machine;

another machine automation server object adapted to execute in the server process;

a second machine identifier received by said another machine automation server object identifying another client machine; and

another machine automation client object identified by the second machine identifier and adapted to execute on said another client machine in communication with said another machine automation server object via a communications mechanism.

16. (Previously Presented) A method for automating control of a client machine under control of a server process, the method comprising:

executing a machine automation control module in the server process;

instantiating a machine automation server object of a predefined machine automation server object class in the server process, under command of the machine automation control module;

instructing the machine automation server object to instantiate a machine automation client object of a predefined machine automation client object class on the client machine;

instructing the machine automation server object to cause the machine automation client object to reboot the client machine; and

re-establishing communications between the machine automation server object and the machine automation client object via a communications mechanism after rebooting of the client machine completes.

17. (Previously Presented) The method of claim 16 further comprising:

providing the machine automation server object with an identifier of the client machine on which the machine automation client object is to be instantiated; and

instantiating the machine automation client object on the client machine specified by the identifier.

18. Cancelled

19. (Previously Presented) The method of claim 16 further comprising:

restoring a value of a property of the machine automation client object after rebooting of the client machine completes.

20. (Previously Presented) The method of claim 16 further comprising:

executing a first instruction of the machine automation control module that instructs the machine automation server object to cause the machine automation client object to reboot the client machine; and

delaying execution of a subsequent instruction of the machine automation control module until after rebooting of the client machine completes.

21. (Previously Presented) The method of claim 20 wherein the delaying operation delays execution of a subsequent instruction of the machine automation control module until after a specified time after rebooting of the client machine completes.

22. (Previously Presented) The method of 16 further comprising:

executing a first instruction of the machine automation control module that instructs the machine automation server object to cause the machine automation client object to reboot the client machine; and

executing a subsequent instruction of the machine automation control module after the machine automation server object causes the machine automation client object to reboot the client machine and before rebooting of the client machine completes.

23. (Previously Presented) The method of claim 16 further comprising:

executing the server process in a server machine;

instructing the machine automation server object to cause the machine automation client object to log off of the client machine; and

instructing the machine automation server object to cause the machine automation client object to re-log in to the client machine using a predetermined user name.

24. (Previously Presented) The method of claim 23 further comprising:

recording the predetermined user name in a system registry of the client machine; and

instructing the machine automation server object to cause the machine automation client object to log into the client machine using the predetermined user name read from the system registry.

25. (Previously Presented) The method of 16 further comprising:

executing a first instruction of the machine automation control module that instructs the machine automation server object to cause the machine automation client object to reboot the client machine; and

executing a subsequent instruction of the machine automation control module after the machine automation server object causes the machine automation client object to reboot the client machine and before rebooting of the client machine completes.

26. (Original) The method of claim 16 further comprising:

executing the server process in a server machine;

instructing the machine automation server object to cause the machine automation client object to reboot the client machine; and

instructing the machine automation server object to cause the machine automation client object to re-log in to the client machine after rebooting completes.

27. (Original) The method of claim 16 further comprising:

instructing the machine automation server object to cause the machine automation client object to access a system registry of the client machine, under command of the machine automation control module.

28. (Original) The method of claim 27 wherein the operation of instructing the machine automation server to cause the machine automation client object to access the system registry comprises:

instructing the machine automation server object to cause the machine automation client object to return to the machine automation server object a snapshot of the system registry of the client machine, under command of the machine automation control module.

29. (Original) The method of claim 16 further comprising:

instructing the machine automation server object to cause the machine automation client object to return to the machine automation server object a snapshot of a file system of the client machine, under command of the machine automation control module.

30. (Original) The method of claim 16 further comprising:

instructing the machine automation server object to cause the machine automation client object to return system information about the client machine to the machine automation server object, under command of the machine automation control module.

31. (Original) The method of claim 16 further comprising:

instructing the machine automation server object to cause the machine automation client object to install an application on the client machine, under command of the machine automation control module.

32. (Previously Presented) The method of claim 16 further comprising:

providing to the machine automation server object a first machine identifier identifying the client machine;

instantiating another machine automation server object of the machine automation server object class in the server process, under command of the machine automation control module;

providing to said another machine automation server object a second machine identifier identifying another client machine;

instructing said another machine automation server object to instantiate another machine automation client object of the machine automation client object class on said another client machine via a communications mechanism; and

instructing said another machine automation server object to cause said another machine automation client object to control operation of said another client machine.

33. Cancelled.

34. (Previously Presented) A computer program storage medium readable by a computer system and encoding a computer program for executing a computer process automating control of a client machine under control of a server process, the computer process comprising:

executing a machine automation control module in the server process;

instantiating a machine automation server object of a predefined machine automation server object class in the server process, under command of the machine automation control module;

instructing the machine automation server object to instantiate a machine automation client object of a predefined machine automation client object class on the client machine;

instructing the machine automation server object to cause the machine automation client object to reboot the client machine; and

re-establishing communications between the machine automation server object and the machine automation client object via a communications mechanism after rebooting of the client machine completes.

35. (Currently Amended) A computer program storage medium readable by a computer system ~~program product~~ and encoding a computer program for executing on a computer system a computer process for automating control of a first client machine and a second client machine under control of a server process via a communications mechanism, for testing software on the first client machine and the second client machine, the computer process comprising:

executing a machine automation control module in the server process;

instantiating a first machine automation server object of a predefined machine automation server object class in the server process, under command of the machine automation control module;

instructing the first machine automation server object to instantiate a first machine automation client object of a predefined machine automation client class on the first client machine, and the first machine automation client object for executing one or more testing methods on the client machine;

instantiating a second machine automation server object of the machine automation server object class in the server process, under command of the machine automation control module;

instructing the second machine automation server object to instantiate a second machine automation client object of the machine automation client object class on the second client machine, and the second machine automation client object for executing one or more testing methods on the client machine;

instructing the first machine automation server object to cause the first machine automation client object to control testing of the first client machine; and

instructing the second machine automation server object to cause the second machine automation client object to control testing of the second client machine.

36. (Previously Presented) The computer program product of claim 35 wherein the computer process further comprises:

executing the server process in a server machine; and

instructing the first machine automation server object to cause the first machine automation client object to reboot the first client machine;

instructing the second machine automation server object to cause the second machine automation client object to reboot the second client machine;

re-establishing communications between the first machine automation server object and the first machine automation client object via the communications mechanism after rebooting of the first client machine completes; and

re-establishing communications between the second machine automation server object and the second machine automation client object via the communications mechanism after rebooting of the second client machine completes.

37. (Original) The computer program product of claim 36 wherein the computer process further comprises:

rebooting the first client machine, responsive to the operation of instructing the first machine automation server object to cause the first machine automation client object to reboot the first client machine; and

rebooting the second client machine concurrently with the operation of rebooting the first client machine, responsive to the operation of instructing the second machine automation server object to cause the second machine automation client object to reboot the second client machine.